

We Claim:

1 1. A graphics system including:

2 a producer that outputs graphics commands,

3 a consumer that consumes the graphics commands outputted by the producer,

4 and

5 a storage device coupled between the producer and the consumer, the storage
6 device storing at least one buffer that receives and temporarily stores graphics
7 commands outputted by the producer for delivery to the consumer, wherein the
8 producer and the consumer are capable of accessing said buffer independently of
9 one another.

1 2. A graphics system as in claim 1 wherein the producer and the consumer

2 have independent read and/or write pointers.

1 3. A graphics system as in claim 1 wherein the storage device stores plural

2 variable sized buffers disposed at selected locations within the storage device.

1 4. A graphics system as in claim 1 wherein the storage device stores plural

2 buffers each of which can be independently accessed by the producer and/or the
3 consumer.

1 5. The graphics system of claim 4 wherein the consumer is incapable of

2 writing to at least an active one of the plural buffers, but maintains, independently of
3 the producer, a write pointer for at least said active one of the plural buffers.

1 6. The graphics system of claim 2 wherein the consumer selectively

2 increments the consumer write pointer in response to the producer writing to the
3 active buffer.

1 7. The graphics system of claim 4 wherein the producer provides a producer
2 read pointer and a producer write pointer associated with a first of said plural
3 buffers, and the consumer independently maintains a consumer read pointer and a
4 consumer write pointer associated with said first of said plural buffers.

1 8. The graphics system of claim 6 wherein the consumer increments the
2 consumer read pointer as the consumer reads from an active buffer, and suspends
3 reading from the active buffer when the incremented consumer read pointer has a
4 predetermined relationship with the consumer write pointer.

1 9. The graphics system of claim 4 wherein a first of the plural buffers
2 includes a read command controlling the consumer to consume a set of graphics
3 commands the producer stores elsewhere within the storage device, and to resume
4 consuming graphics commands from the first buffer after consuming the graphics
5 commands stored elsewhere.

1 10. The graphics system of claim 9 wherein the read command specifies a
2 starting address and a length of a display list, the read command controlling the
3 consumer to read the display list of the specified length beginning at the specified
4 starting address.

1 11. The graphics system of claim 1 wherein the buffer provides circular first-
2 in-first-out access.

1 12. The graphics system of claim 1 wherein the buffer provides linear first-
2 in-first-out access.

1 13. The graphics system of claim 1 wherein the buffer can be selectively
2 attached to both the producer and the consumer simultaneously.

1 14. The graphics system of claim 1 wherein the buffer is attached to the
2 producer while a further buffer is attached to the consumer.

1 15. The system of claim 1 wherein the buffer is first attached to the producer,
2 and is subsequently detached from the producer and attached to the consumer.

1 16. The system of claim 1 wherein the buffer can be attached to the
2 producer, the consumer, or both.

1 17. The system of claim 1 wherein only one buffer is attached to the
2 producer at a time.

1 18. The system of claim 1 wherein only one buffer is attached to the
2 consumer at a time.

1 19. The system of claim 1 wherein the buffer has a maximum size of 16 KB.

1 20. The system of claim 1 wherein the producer sets the size of the buffer.

1 21. The system of claim 1 wherein the buffer is dynamically sized to store a
2 frame of graphics commands.

1 22. The system of claim 1 wherein the producer declares the buffer by
2 issuing a graphics buffer initialization command specifying buffer starting address
3 and buffer length.

1 23. The system of claim 1 wherein the buffer has a length that is a multiple of
2 32 bytes and has a minimum size of 64 KB.

1 24. The graphics system of claim 1 wherein the producer may write a
2 breakpoint into the buffer, the consumer suspending consumption of graphics
3 commands upon encountering the breakpoint.

1 25. The system of claim 1 wherein the buffer has an overflow status indicator
2 indicating when the producer overwrites a location therein.

1 26. The system of claim 1 further including a hardware status register that
2 indicates the status of the buffer.

1 27. The system of claim 26 wherein the status register includes the following
2 parameters:

3 position of a producer write pointer relative to buffer full and buffer empty;

4 buffer overflow;

5 whether the producer is currently writing into the buffer; and

6 whether the consumer is currently reading from the buffer.

1 28. The system of claim 1 further including a hardware controller coupled to
2 the buffer, the hardware controller providing flow control logic to prevent writes
3 from overrunning reads.

1 29. The system of claim 1 further including a hardware controller coupled to
2 the buffer, the hardware controller wrapping read and write pointers from a last
3 location to a first location thereof.

1 30. The graphics system of claim 1 wherein the producer comprises a
2 processor and the consumer comprises a graphics processor including a graphics
3 pipeline.

1 31. The system of claim 1 wherein the storage device comprises a main
2 system memory, and the producer dynamically allocates said buffer within the main
3 system memory.

1 32. A graphics system including:

2 a storage buffer that receives and temporarily stores graphics commands;
3 a producer that writes graphics commands into said buffer, said producer
4 maintaining a producer write pointer and a producer read pointer associated with the
5 buffer; and
6 a consumer that consumes the graphics commands stored within the buffer,
7 the consumer maintaining a consumer write pointer that is independent of the
8 producer write pointer, and a consumer read pointer that is independent of the
9 producer read pointer.

1 33. The graphics system of claim 32 wherein the consumer increments the
2 consumer read pointer each time the consumer consumes from the buffer, and
3 suspends consumption from the buffer when the consumer read pointer has a
4 predetermined relationship with the consumer write pointer.

1 34. The graphics system of claim 32 wherein the consumer selectively auto-
2 increments the consumer write pointer in response to the producer writing to the
3 buffer.

1 35. The graphics system of claim 32 wherein the producer sends the
2 consumer a configuration command specifying whether the consumer should auto-
3 increment the consumer write pointer in response to producer writes to the buffer.

1 36. In a graphics system including a graphics command producer that writes
2 graphics commands into a buffer based on a producer write pointer and a graphics
3 command consumer that reads graphics commands from the buffer based on a
4 consumer read pointer,
5 an improvement comprising:

6 a consumer write pointer independently maintained by the consumer, the
7 consumer write pointer indicating the extent of valid data the producer has written
8 into said buffer, the consumer ceasing to consume graphics commands from the
9 buffer upon the consumer read pointer having a predetermined relationship to the
10 consumer write pointer.

1 37. In an interactive graphics system including a processor module executing
2 an application, a graphics processor module and at least one memory coupled to the
3 processor module and to the graphics processor module, a method of controlling the
4 flow of graphics commands between the processor module and the graphics
5 processor module comprising:

6 dynamically establishing, under control of the application, a variable number
7 of FIFO buffers in the memory, the application specifying the size of each of the
8 FIFO buffers;

9 the application controlling the processor module to write graphics commands
10 into at least a first of the plurality of FIFO buffers; and

11 the application sending graphics commands to the graphics processor module
12 that control the graphics processor module to read independently of said processor
13 writes, the graphics command from the first FIFO buffer.

1 38. The graphics system of claim 37 wherein the processor module provides
2 a processor module read pointer and a processor module write pointer associated
3 with a first of said plurality of buffers, and the graphics processor module
4 independently maintains a graphics processor module read pointer and a graphics
5 processor module write pointer associated with said first buffer.

1 39. The graphics system of claim 37 wherein the graphics processor module
2 increments the graphics processor module read pointer each time the graphics
3 processor module reads from the first buffer, and suspends reading from the first
4 buffer when the graphics processor module read pointer has a predetermined
5 relationship with the graphics processor module write pointer.

1 40. The graphics system of claim 37 wherein the graphics processor module
2 selectively auto-increments the graphics processor module write pointer in response
3 to the processor module writing to the first buffer.

1 41. The graphics system of claim 37 wherein the graphics processor module
2 maintains, independently of the processor module, a write pointer for at least an
3 active one of the plurality of buffers.

1 42. The method of claim 37 further including setting a breakpoint, and at
2 least temporarily suspending the graphics processor module from reading a buffer in
3 response to the graphics processor module encountering the breakpoint.

1 43. The method of claim 37 wherein the selectively controlling step includes
2 suspending the processor module from writing to the buffer upon detection of an
3 overflow.

1 44. A method of controlling the flow of graphics data comprising:
2 writing graphics data into plural variable sized FIFO buffers each having
3 plural storage locations;
4 setting a breakpoint associated with at least one of the plural storage
5 locations;
6 reading graphics data from the plural buffers in a predetermined order;

7 temporarily suspending the reading step upon encountering the at least one
8 location associated with the breakpoint, and generating an interrupt; and
9 resuming the reading step in response to receipt of a clear interrupt command.

1 45. The method of claim 44 wherein the resuming step is performed in
2 response to receipt of a disable breakpoint command.

1 46. The method of claim 44 wherein:

2 the writing step comprises writing graphics data corresponding to a first
3 image frame into a first part of the buffer, and writing graphics data corresponding
4 to a second image frame into a second part of the buffer; and

5 the setting step comprises setting the breakpoint to be associated with a
6 location separating the first and second buffer portions.

1 47. The method of claim 44 wherein the predetermined order is first-in-first-
2 out.

1 48. The method of claim 44 wherein the writing step is performed by general
2 purpose processing hardware, and the reading step is performed by special purpose
3 graphics hardware.

1 49. The method of claim 44 wherein the reading and writing steps are
2 performed substantially simultaneously.

1 50. The method of claim 44 wherein the writing step completes before the
2 reading step begins.

1 51. The method of claim 44 wherein the resumed reading step continues until
2 a further breakpoint is encountered, the buffer is empty, or an image frame
3 represented by the graphics data is aborted.

1 52. The method of claim 44 further including indicating the status of the
2 buffer.

1 53. The method of claim 52 wherein the status indicating step includes
2 indicating at least three of the following parameters:

3 position of a write pointer relative to buffer full and buffer empty;

4 buffer overflow;

5 the writing step is active;

6 whether command processing is idle; and

7 the reading step is active.

1 54. The method of claim 44 wherein the writing step comprises writing the
2 buffer circularly, and the reading step comprises reading the buffer circularly.

1 55. A graphics system including:

2 a storage device that receives and temporarily stores graphics commands;

3 a producer that writes commands into a buffer within said storage device, said
4 commands including a first set of graphics commands and a command referencing a
5 second set of graphics commands stored elsewhere within said storage device; and

6 a consumer that consumes the first set of graphics commands stored within
7 the buffer and, in response to encountering the referencing command, consumes the
8 second set of graphics commands referenced thereby and subsequently returns to the
9 buffer to consume additional commands therefrom.

1 56. The graphics system of claim 55 wherein the buffer is a circular buffer.

1 57. The graphics system of claim 55 wherein the referencing command
2 specifies a starting address of a display list, the referencing command controlling the
3 consumer to read the display list beginning at the specified starting address.

1 58. The graphics system of claim 55 wherein the referencing command
2 specifies a number of data units the consumer is to consume.

1 59. The graphics system of claim 55 wherein the consumer is incapable of
2 writing to the buffer, but maintains, independently of the producer, a write pointer
3 for the buffer.

1 60. In a graphics system, a method for passing graphics commands from a
2 producer of graphics commands to a consumer of graphics commands, the method
3 comprising:

4 creating plural variable sized buffers disposed at variable locations within a
5 memory coupled to the producer and the consumer,

6 temporarily storing graphics commands produced by the producer in the
7 variable sized buffers;

8 consuming the graphics commands from the variable sized buffers with the
9 consumer by accessing the buffers independently of the producer; and

10 generating at least a part of a graphics image based at least in part on the
11 consumed graphics commands.

1 61. The method of claim 60 wherein the consumer is incapable of writing to
2 at least an active one of the plural buffers, and the method further includes the
3 consumer maintaining, independently of the producer, a write pointer for at least
4 said active one of the plural buffers.

1 62. The method of claim 60 further including maintaining, with the producer,
2 a producer read pointer and a producer write pointer associated with a first of said
3 plural buffers, and the independently maintaining, with the consumer, a consumer
4 read pointer and a consumer write pointer associated with said first of said plural
5 buffers.

1 63. The graphics system of claim 62 further including incrementing, with the
2 consumer, a consumer read pointer as the consumer reads from an active buffer, and
3 suspending reading from the active buffer when the incremented consumer read
4 pointer has a predetermined relationship with the consumer write pointer.

1 64. The graphics system of claim 63 further including selectively
2 incrementing the consumer write pointer in response to the producer writing to the
3 active buffer.

1 65. The method of claim 60 wherein a first of the plural buffers includes a
2 read command, and the method further includes:

3 (a) consuming a set of graphics commands the producer stores elsewhere
4 within the storage device in response to encountering the read command, and

5 (b) resuming consumption of graphics commands from the first buffer after
6 consuming the graphics commands stored elsewhere.

1 66. The graphics system of claim 65 wherein the read command specifies a
2 starting address and a length of a display list, and step (a) includes controlling the
3 consumer to read the display list of the specified length beginning at the specified
4 starting address.

1 67. The method of claim 60 wherein a first of the plural buffers provides
2 circular first-in-first-out access.

1 68. The method of claim 60 wherein a first of the plural buffers provides
2 linear first-in-first-out access.

1 69. The method of claim 60 further including selectively attaching any of the
2 plural buffers to both the producer and the consumer simultaneously.

1 70. The method of claim 60 further including attaching a first of the plural
2 buffers to the producer and attaching a second of the plural buffers to the consumer.

1 71. The system of claim 60 further including attaching a first of the plural
2 buffers to the producer, and subsequently detaching the first buffer from the
3 producer and attaching the first buffer to the consumer.

1 72. The system of claim 60 further including attaching any of the plural
2 buffers to the producer, the consumer, or both.

1 73. The system of claim 60 further including attaching only one of the plural
2 buffers to the producer at a time.

1 74. The system of claim 60 further including attaching only one of the plural
2 buffers to the consumer at a time.

1 75. A method for producing images including:

2 maintaining a producer write pointer and a producer read pointer associated
3 with a buffer;

4 writing graphics commands into the buffer, and updating at least the write
5 pointer in response to the writing;

6 maintaining, in association with the buffer, a consumer write pointer that is
7 independent of the producer write pointer, and a consumer read pointer that is
8 independent of the producer read pointer;

9 consuming the graphics commands stored within the buffer, and updating at
10 least the read pointer in response to the consuming; and

11 producing at least a part of a graphics image at least in part in response to the
12 consuming step.

1 76. A method of producing images including:

2 writing commands into a buffer within a storage device, said commands
3 including a first set of graphics commands and a command referring to a second set
4 of graphics commands stored elsewhere within said storage device;

5 consuming the first set of graphics commands stored within the buffer;

6 in response to encountering the referring command, consuming the second set
7 of graphics commands and subsequently automatically returning to consume
8 additional commands from the buffer; and

9 generating at least a part of an image at least in part in response to the
10 consumed first and second sets of graphics commands.

1 77. The method of claim 76 wherein the second set of graphics commands
2 comprises a display list.

1 78. A method of generating data structures comprising:

2 writing a graphics command stream into a command stream buffer, said
3 command stream including a predetermined command signifying the beginning of a
4 display list; and

5 in response to the predetermined command, redirecting the graphics
6 command stream from the command stream buffer to a display list buffer.

1 79. The method of claim 78 wherein the command stream buffer comprises a
2 first-in-first-out buffer.

1 80. The method of claim 78 further including writing a further predetermined
2 command that redirects the graphics command stream from the display list buffer
3 back to the command stream buffer.

1 81. A method of supplying 3D graphics commands to a 3D graphics
2 command consumer comprising:

3 (a) storing a command sequence beginning at a predetermined storage
4 location; and

5 (b) supplying a graphics command stream through a FIFO buffer to the
6 consumer, the stream including at least one command that refers the command
7 consumer to the predetermined storage location,

8 wherein the producer returns to the FIFO buffer after consuming the
9 command sequence beginning at the predetermined storage location.